Available online at www.sciencedirect.com
ScienceDirect
Journal of Biomedical Informatics 40 (2007) 672–687
Journal of Biomedical Informatics
www.elsevier.com/locate/yjbin

Knowledge construction from time series data using a collaborative exploration system

Thomas Guyet [a,*], Catherine Garbay [b], Michel Dojat [c,d]

[a] CNRS—TIMC/LIG—Grenoble, France
[b] CNRS—LIG—Grenoble, France
[c] Inserm, U836, Grenoble, France
[d] Université Joseph Fourier, Institut des Neurosciences, Grenoble, France

Received 22 September 2006
Available online 9 October 2007

Abstract

This paper deals with the exploration of biomedical multivariate time series to construct typical parameter evolution or scenarios. This task is known to be difficult: the temporal and multivariate nature of the data at hand and the context-sensitive aspect of data interpretation hamper the formulation of *a priori* knowledge about the kind of patterns that can be detected as well as their interrelations. This paper proposes a new way to tackle this problem based on a human–computer collaborative approach involving specific annotations. Three grounding principles, namely autonomy, adaptability and emergence, support the co-construction of successive abstraction levels for data interpretation. An agent-based design is proposed to support these principles. Preliminary results in a clinical context are presented to support our proposal. A comparison with two well-known time series exploration tools is furthermore performed.

Keywords: Human/Machine interface; Collaborative systems; Data interpretation; Intensive care data; Multi-agent system; Knowledge discovery

1. Introduction

Making life-critical decisions based on multivariate time series data is a relatively common task in medical domains. This task turns out to be of particular importance in the context of ICU (Intensive Care Unit) patient monitoring. As rather commonly recognized, the large mass of data that is made available to ICU staff may not be fully exploited due to a lack of time, a lack of staff, and a lack of formalized knowledge. The need for computer assistance in this specific domain has already been pointed out and is widely admitted.

Considering the lack of formalized knowledge to design such a computerized assistant, we rather explore the potential of a man–machine collaborative approach as a preliminary step. The goal is to extract chunks of knowledge that will bring a better understanding of the data at hand and support forthcoming decision making process. Data exploration is therefore required. It consists of (1) the focus of the attention on the segments of interest, (2) the extraction of significant patterns, (3) the combination of them to construct meaningful scenarios, and finally (4) the verification of the adequacy and significance of the produced chunks of knowledge.

Considering the combinatorial character of these tasks, an incremental approach is proposed in which both, human and machine, collaborate toward the progressive exploration of the data at hand and the gradual construction and refinement of the data processing models. Annotations are advocated as an elegant and efficient way of communication between human and machine. They may be provided, at various abstraction levels, in the form of segment delineations or symbolic labeling. Starting with a limited range of manually provided annotations, the system is meant to build its own models, delineate new segments, label new patterns, and thus further annotate the data. The clinician may intervene at any time during this process to provide further annotations, or modify current segment delineation and labeling.

* Corresponding author. Fax: +33 4 56 52 00 22.
E-mail address: Thomas.Guyet@imag.fr (T. Guyet).

1532-0464/$ - see front matter © 2007 Elsevier Inc. All rights reserved.
doi:10.1016/j.jbi.2007.09.006

Our design is grounded on three main principles, namely autonomy (ability to learn), adaptability (ability to cope with the pattern variability and heterogeneity of features associated with each class), and emergence (co-evolution of the models and annotations in the course of man–machine collaboration). An agent-centered design is proposed as a way to computerize these principles.

The structure of the paper is as follows. Section 2 is devoted to a brief state of the art on collaborative knowledge construction and data exploration. The proposed collaborative approach is described in Section 3 and the multi-agent architecture is presented in Section 4. Experimental results on ventilation asynchronies are presented in Section 5 and followed by a discussion in Section 6.

2. State of the art

2.1. Issues in knowledge acquisition

Since the early work on expert system design, human knowledge acquisition has remained a critical but open issue. Research efforts have rapidly been devoted to data-driven knowledge extraction to cope with the growing need for mass treatment. In most fields of medicine the number of variables is indeed increasing in a way which precludes medical judgment by humans, as pointed out by [1]. This statement especially holds in the field of ICU, one of the most data intensive environments in medicine [2]. In [2] a comparison between data-driven temporal abstraction, which exploits quantitative featuring in the framework of knowledge discovery techniques, and knowledge-driven abstraction, which exploits expert knowledge expressed in qualitative form, has been conducted. Both approaches have been evaluated on their capacity to predict whether post-surgical patients would need mechanical ventilation for longer than 24 h. The data-driven approach has been found to provide more informative cues, thus resulting in better predictions.

More recently, various attempts have been made to articulate quantitative measurements with a more qualitative style of reasoning. The necessity to mix data-centered analysis with human-driven reasoning has been widely recognized. Nowadays, data mining is increasingly considered as a cooperative process relying on background knowledge to drive data exploration; the means to integrate the newly acquired findings within the expert knowledge are furthermore increasingly considered. Zupan et al. [3] propose the notion of ''knowledge circle'' to formalize the necessary alternation between knowledge-driven data analysis and data-driven knowledge construction. As mentioned by the authors, the automation of this cycle, although highly desired, is rarely achieved in real-world applications.

2.2. Collaborative knowledge construction

The notion of ''Balanced Cooperative Modelling'' was early proposed by Morik et al. [4], as a multi-strategy approach to domain modelling. A balanced interaction between system and user was proposed to enrich the domain knowledge. In further work [1], the authors present an approach involving intelligent data analysis together with knowledge acquisition from experts to support the development of operational protocols in the intensive care field. The guiding principle aims at enriching the knowledge available by mixing various knowledge sources. In addition, the authors emphasize the gain in robustness that may be obtained by ''cross-validating'' the knowledge sources: while human knowledge may gain from a confrontation to machine-driven constructs, the necessity to incorporate newly discovered findings into already formalized expert knowledge functions as a form of validation.

Going a step further, Shroeder and Bazzan [5] propose a combination of learning algorithms in a multi-agent framework to improve individual models through knowledge sharing. This environment, called MASKS (Multi-Agent System based on Knowledge Sharing) proved to be efficient when applied in bio-informatics. Each agent which is derived of machine learning algorithm generates a set of rules. The cooperative learning, based on pair wised interactions, improves the quality of already established rules. Two agents may match or merge their rules by comparing their classification results and exchange their models. The system outperforms the results obtained by a single machine learning algorithm and thus indicates that collaboration may support efficient knowledge discovery.

2.3. Collaborative data exploration

Collaborative data exploration is a complement to the collaborative knowledge construction in the development of rich ''knowledge cycles''. Indeed, while there must be ways to integrate machine discovered findings in the realm of human conceptual knowledge, there must be ways for the human to check the validity of his/her models against the data at hand. Collaborative data exploration is thus a way to investigate complex scenes, to suggest changes in attention focus, to question the proposed modeling, or to test new hypotheses. The knowledge circle may then be fully approached by mixing the exploration at the machine initiative (the integration of machine discovered findings within the realm of human concepts) and the exploration at the human initiative with the confrontation of human knowledge against the realm of data.

Several tools have been proposed such as BinX [6], QuerySketch [7] or TimeTunnel [8], providing support for the interactive visualization and exploration of time-oriented clinical data. Going a step further towards the temporal abstractions handling, the driving view in KNAVE and then KNAVE II [9] was to develop a process able to embody domain-independent abstraction methods while exploiting domain-specific temporal-abstraction knowledge.

Recent research efforts have been devoted to the integration of knowledge discovery tools to support in-depth exploration of time series. VizTree [10] has been designed as a pattern discovery and visualization system able to summarize both the global and local structures of time series data. It is based on SAX [11], a symbolic representation for time series, and on a modified suffix tree to encode the properties of data. VizTree provides a rather intuitive way to interact with the input data and the corresponding structures. It provides novel interactive solutions to many pattern discovery problems, such as motif discovery, anomaly detection, and query by content. TimeSearcher2 [12] is rather oriented towards the interactive formulation and modification of user-defined queries. This is achieved by so-called “TimeBoxes”, a kind of graphical widget by which the user may point out different patterns of interest. Conjunctive or disjunctive queries may further be expressed as the combination of multiple query items. Utilizing this mechanism users may look for time series exhibiting specific patterns as specified in the query.

3. Methodological approach

3.1. Time series interpretation

Temporal data abstraction is known to be a core issue in medicine, and there is a wide literature tackling this issue in various medical domains [13]. Such processing still remains challenging, due to the necessity to consider compound objects (e.g. disorders, treatments, or patient states). These exhibit different temporal existences and complex interactions through mechanisms that are not completely understood. The complexity of this process has been particularly well addressed by Shahar [14], who quoted the necessity to articulate several mechanisms operating at various abstraction levels and grounded on explicit knowledge. The study of specific mechanisms goes beyond the scope of this paper, therefore we focus in this section on an overview of our basic assumptions.

We consider time series interpretation as a mere abstraction process by which more abstract annotations are progressively elaborated and attached to the raw data. In our system, several steps are necessary before reaching the most abstract annotation, i.e. the scenarios:

- *Segmentation:* The goal of this step is to achieve a preliminary description of the data as a set of temporal meaningful segments. These segments are detected independently for each time series.
- *Symbolic time series transformation:* The goal of this step is to transform the signal-level information into symbolic form. Each segment in each time series is given a symbolic “name”. A symbol is related to models of events. Successive segments are then concatenated to constitute symbolic time series, in which the temporal information is preserved.
- *Scenario construction:* A set of frequent time-stamped symbols is extracted from the symbolic time series, together with their temporal constraints in order to build scenarios from the available multivariate information.

This abstraction process, when not carefully performed, is known to result into so-called “semantic gap” problems. In such a case the proposed symbolic information derives from a rather high level conceptual view on the problem at hand, rather than reflecting the effective properties of the data under consideration. In other words, in a medical context it is difficult for a clinician to relate his or her knowledge to the processing style of the machine.

Four major assumptions are made in this respect:

- *A priori* knowledge is needed at each processing step, whatever its abstraction level, but sparse and difficult to acquire;
- Knowledge must be anchored in the realm of data, i.e. situated, rather than relying on global modeling hypotheses;
- An incremental processing style is necessary to refine the analysis and accumulate progressively more robust knowledge;
- Augmented styles of collaboration between man and machine are needed to ensure a proper formulation and transmission of knowledge.

3.2. Human–machine collaboration

We propose to approach the interpretation of time series data as performed by two cooperating agents—a man and a machine—operating across three successive abstraction levels (segments, symbolic time series, and scenarios). Considering that these are complementary, this collaboration is necessary to cope with the complexity of the interpretation process. The clinician is recognized as able to take accurate decisions in rather complex situations, by integrating a wide range of contextual information and keeping a global outlook over the data at hand. Conversely, the machine is able to process large amounts of data under complex numerical constraints.

The collaboration or cooperation “occurs when two or more agents work together in a common environment to more effectively reach the maximal union of their goal” [15]. In [16], the authors define the main characteristics expected for man–machine collaboration system based on the study of man–man cooperation. Following these authors, the system must:

- Operate within an acceptable framework of coordination.
- Be able to recognize and accept the collaborator’s goals when declared.
- Be able to interactively work toward super ordinate goals in solving complex tasks.

– Offer alternative solutions to the problem addressed.
– Operate to support the formation of new attitudes (adaptation).

Although considering the collaboration between men as a model for man–machine collaboration can be criticized [17], it gives us a frame of reference for collaborative approaches.

The mixed-initiative approach [18] appears to date as the more elaborated tentative to achieve these characteristics. In a mixed-initiative system both the system and the user have balanced contributions for problem solving. A synergy between the two agents is expected to fruitfully integrate complementary abilities to be globally more efficient. In [19], an assistant for exploratory data analysis has been developed based on this approach.

In the same vein, we propose an approach based on a "structural coupling" [20] between man and machine to reach collaboration. The man and machine "may become reciprocally structurally coupled through their reciprocal selection of plastic structural changes during their history of interactions. In such a case, the structurally plastic changes of state of one system become perturbations for the other, and vice versa, in a manner that establishes an interlocked, mutually selecting, mutually triggering domain of state trajectories" [21]. We point out three differences between a mixed-initiative approach and the proposed structural coupling approach:

– The role of our system is not to recognize the human needs in order to assist her/him, but to contribute in a balanced way to the solution.
– In our system man and machine share a common goal.
– The adaptation during the problem solving process is central in the structural coupling approach.

In a way similar to the talking heads of Steels [22], which interact to build a shared lexicon, based on their independent perception and analysis of geometrical figures, we propose to consider both the clinician and the machine as agents who share a common environment (e.g. time series data, segments, scenarios, etc.), and mutually interact to progressively refine their interpretation.

In the absence of consistent *a priori* knowledge and considering the degree of difficulty of this process, an active partnership between man and machine is to be sought. It does not only consist of a fixed request–answer interaction scheme in which each partner is meant to compensate for the lack of knowledge of its partner, and therefore supposed to share its world of meaning. Based on [22,23], this partnership is rather meant to allow a co-construction of meaning, in which the interpretation of facts is not defined beforehand by one of the partners, but co-constructed in the course of their interactions.

Annotation [24] is a core concept to cope with these difficult issues. It receives a growing interest in the co-design field, and has been shown to support the dynamics of co-operation. Annotations may be seen as tangible marks that can be managed by the partners, i.e. they enable the co-construction of objects. They may also be seen as tangible signs that make sense, i.e. they are the materialization of contextual knowledge that may be shared among the partners. According to this principle, each partner is in turn given the possibility to observe and interpret annotations provided by its partner, and/or to propose annotations judged as appropriate according to a given interpretation focus. In such a framework, each partner is meant to reason in its own world of meaning, thus preserving its autonomy. Interpretation is not considered as a context-free attribution of meaning, but as grounded in each partner's experience. There is no prevalence of one partner over another, rather, there is a possibility of learning and discovery for both partners.

3.3. Proposed design

Based on the previous assumptions, we define interpretation as a complex process involving complementary exploration, description, and annotation tasks. These tasks are operated by a collection of autonomous agents (man or machine), who work in collaboration to progressively refine the annotation [25]. The role of the exploration task is to focus the attention, at each abstraction level, on relevant information, in terms of time series segments, event classes or typical event relations. The description and annotation tasks are situated in a dynamic context which includes the agent past experience and external information on data. The description task role is to build numerical and symbolical models of the information in the interest to fuse relevant information. Finally, the annotation task role is to annotate new raw data using the constructed models.

As we consider that there is neither a grammatical nor semantical representation of the world available beforehand, interaction has to take place across several abstraction levels. Therefore, the so-called "language game" [22] occurs at the three successive levels of time series segmentation, symbolic transformation, and scenario construction. In the course of the interaction process, each agent proposes in turn its own interpretation and/or points out instances for a given interpretation, according to the following typical scenario:

– *"Machine" side:* Segmentation, symbolization, and scenario construction results are proposed to the human.
– *"User" side:* The user reacts asynchronously (at any time and any processing level) by annotating the proposed interpretation, i.e. inserting for instance a different view on segmentation. She/he may also point out further examples of a given concept, according to its own view.
– *"Machine" side:* In response to the proposed user annotation, the machine may learn new ways for data interpretation and suggest further analysis. It may also point out further examples for a given concept, according to some similarity measure.

It is important to note that that any information—be it a case example or a conceptual interpretation—is provided within its context of appearance, so that interaction develops in a situated way.

An overview of the proposed design is provided in Fig. 1.

4. System design

We present in this section the Multi-Agent System (MAS) that we have designed for collaboratively construct annotations from multivariate time series. From the previous considerations, we derive two major properties for the proposed architecture: (1) It is based on components operating at different abstraction levels and (2) Each of these components is assumed to possess autonomous annotation and learning capabilities.

Time series annotation is considered as an abstraction process involving three separate levels (raw data analysis, symbolic description and scenario construction). Central to our design is therefore the distinction between three main tasks, namely segmentation (partitioning of the signal into consecutive segments), classification (symbolic labeling of the segments), and scenario construction (computation of inter-symbol relationships).

Dedicated agents are conceived to support these various tasks. Before describing these agents, we first of all give an insight into the global system architecture.

4.1. System architecture

A global view of the system architecture is presented in Fig. 2. A triadic organization is proposed as a conceptual framework to formalize agent interactions and feedback loops. A triad is made of two processing entities working in reciprocal interaction on the basis of a more abstract processing whole [26]. Each abstraction level is organized into a triad, thus generating new annotation elements to be processed at a higher abstraction level (upper triad):

- Time series data are processed within the segmentation and classification triads, which mutually interact (links 1 and 2, Fig. 2) within the symbolic translation triad to build symbolic time series (link 3, Fig. 2).
- Symbolic time series are processed within the scenario construction triad. The symbolic translation and scenario construction triads mutually interact within the system triad (links 7 and 8, Fig. 2) to refine and improve the proposed interpretation.
- Scenarios are the final outcome to the system triad (link 9, Fig. 2). Scenarios are used to elaborate modifications to the symbolic time series that might improve scenario accuracy. These modifications are transmitted to the symbolic translation triad (link 7, Fig. 2).

A summary of the annotation elements exchanged between triads is provided in Table 1.

Note that two styles of interaction are considered in this approach to support emergence: the man–machine interaction involved in the segmentation, classification and scenario construction triads, on the one hand, and the triad-to-triad interaction involved in the symbolic translation and system triads on the other hand. Whereas man–machine interaction allows to model the relations occurring between a subject observing an object (ubiquitous interactions), the triad-to-triad interaction allows to model interactions occurring between actors operating at different abstraction levels. According to this view, segmentation for example is meant to occur from the ubiquitous interaction between man and machine, explained as follows.

4.2. Agentification

4.2.1. Segmentation agents

For each patient, a record is provided in the form of a multivariate time series, where each time series is identified by its type. Two data types are considered in our application to ICU patient monitoring, namely the airway pressure (Paw) and respiratory flow signals.

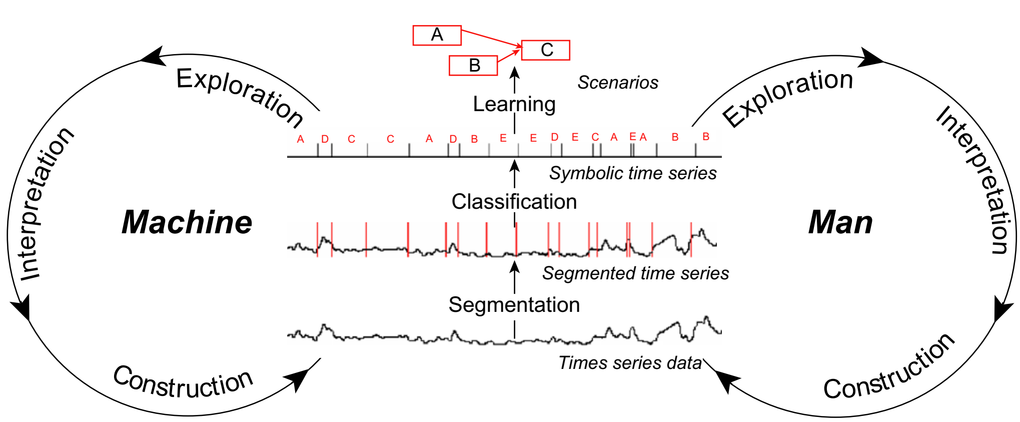

Fig. 1. Time series interpretation as a collaborative process, involving complementary exploration, description, and annotation tasks, operating over several abstraction levels.

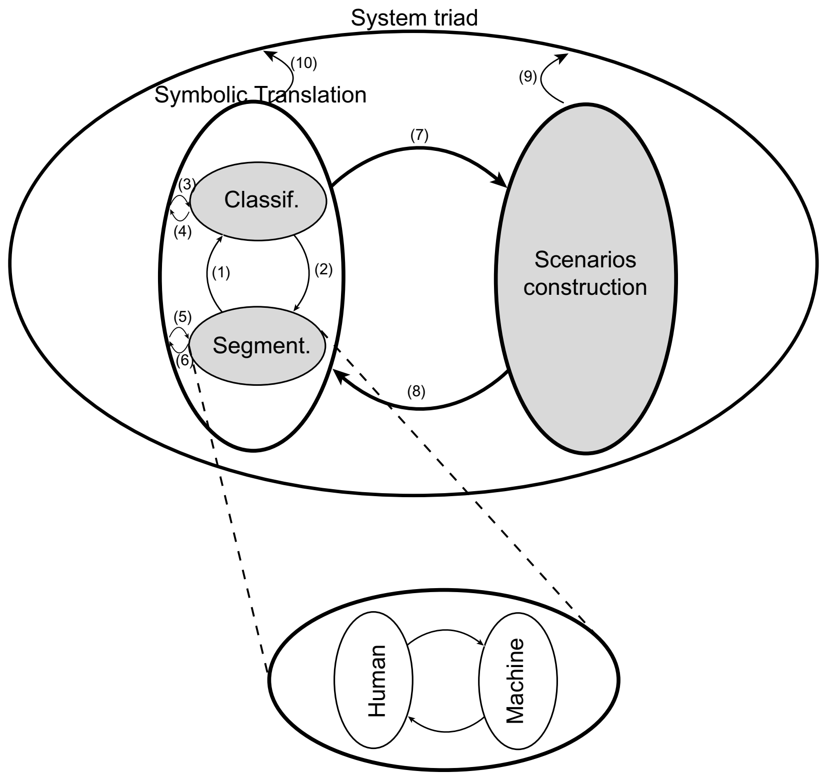

Fig. 2. A view of the system architecture. The system is made of agents organized into recursive triads. The agents mutually interact according to forward and backward links. The segmentation triad is expanded to show the man–machine interaction.

Table 1
Operational descriptions of the MAS triads

Annotation elements triads	From the lower triad	To the opposite triad	To/from the upper triad
Segmentation		(1) Time series partitions (segments)	(6) Set of segments/(5) Segment modifications
Classification		(2) Models of segments	(4) Set of segment models /(3) Modifications of the symbolic translation of segments
Symbolic translation	(6) Set of segments (4) Set of segment models	(7) Symbolic time series	(10) Time-stamped symbols patterns
Scenario construction		(8) Symbolic time series modifications	(9) Scenarios
System	(10) Time-stamped symbols patterns and (9) Scenarios		

Parenthesized numbers correspond to the labeled links shown in Fig. 2.

Various segmentation agents operate upon the collection of time series data. These agents differ in the data type they process and the kind of models they apply. A new segmentation agent is created each time a new segment model is computed by the classification agent. As a consequence segmentation is performed according to the application of several segmentation agents which operate in parallel over the time series data, involving the potential ubiquitous intervention of a human observer. Each part of data matching to a model is considered as a new segment, and the whole set of segments constitutes the segmentation of the time series data, without concern for the precise agent that was involved in this construction.

The match is performed in the model parameter space, rather than in the time series space, to cope with patterns of different sizes—within a certain size range (e.g. patterns occurring across different time intervals). For this purpose, any model is described as a polygonal line consisting of a constant number of n elements and is characterized by its successive coordinates. Each segment is then approximated with n linear piece-wise elements, and its Euclidean distance to the model is computed in the parameter space. The pattern matching algorithm operates in two steps (see Fig. 3). First of all it calculates at each instantaneous position in a time series and for each possible segment size the distance to the pattern model. A distance matrix is then computed. The most accurate segments are extracted in a second step.

Segmentation agents are in charge of applying these pattern matching algorithms for given specific patterns. By

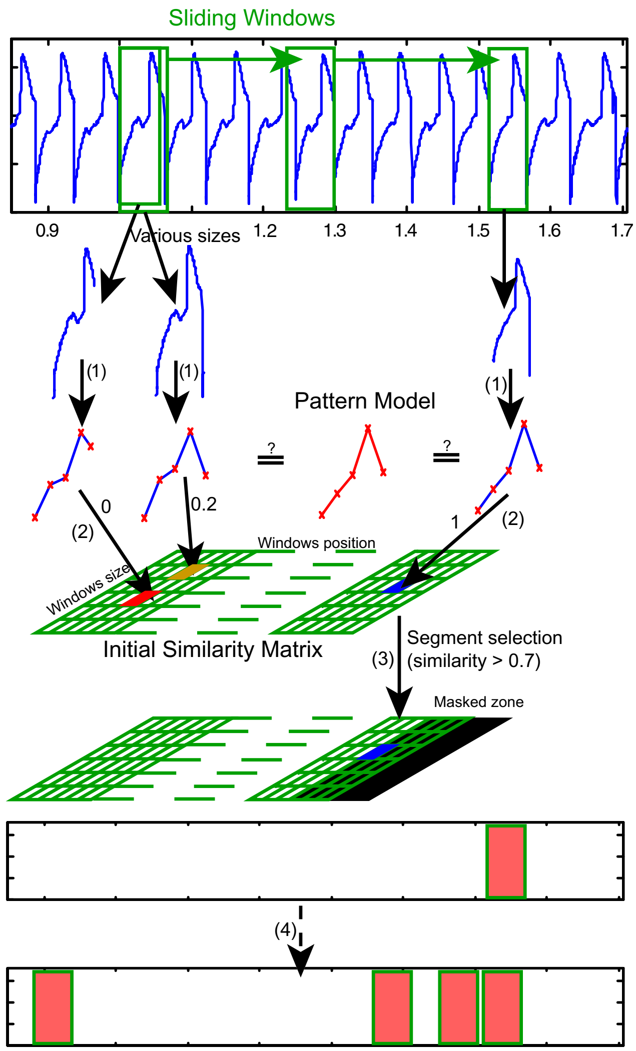

Fig. 3. Pattern matching algorithm. Sliding windows with various sizes are translated over the whole time series. (1) Segment representation in the model space is calculated (here an approximation with five points); (2) Segment to model similarities are computed in the model space as values between 0 and 1; (3) The most accurate segment given a model and a pattern matching threshold (0.7) is selected; Overlapping segment hypotheses are further eliminated (notion of masked zone); (4) The selected segments are collected.

construction a single agent produces non-overlapping segmentations, while a set of agents can operate over different series types and may overlap in time. A non-overlapping constraint is used to eliminate further segment hypotheses: once a segment has been found to match a model, it is masked, so that no overlapping may occur in further matching. Experiments have shown good results for the respiratory parameters time series (see Section 5).

This process is applied independently for each multivariate time series. The results will be fused later on in the course of the process to avoid over-constraining this delicate initial labeling step.

A limited amount of manually annotated segments is provided to the system at starting point to build the initial segment models. Then, annotations are exchanged between man and machine through the interface. Segments are displayed to the user through the system visualization interface. Such visualization allows focusing the user's attention in two ways: while annotated time series elements point out "recognizable" events, non-annotated events conversely either reveal a lack of knowledge, or suggest further exploration of the data.

4.2.2. Classification agents

There is one classification agent per time series type. Every classification agent operates according to the same global process, as illustrated in Fig. 4. The classification process has been designed to involve:

- A rich description of the segments at hand to make the classification process more robust to tiny variations in the signal characteristics.
- An adaptive distance, to make the classification process sensible to individual segment class specificities.
- An incremental algorithm.

A feature vector is computed for each segment that integrates a rich variety of characteristics, e.g. segment width, segment mean value, main trend, or presence of a maxi-

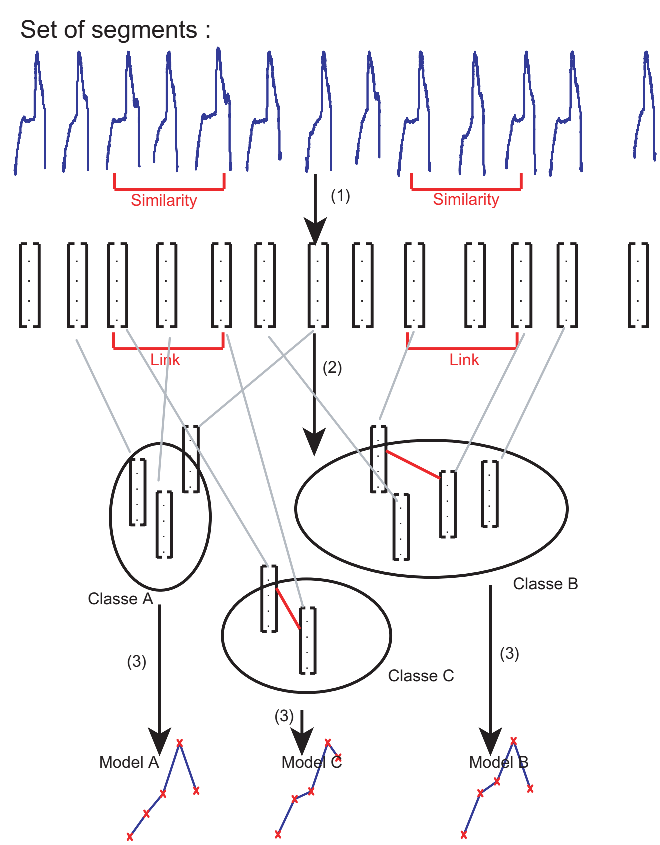

Fig. 4. Segment classification and model generation. (1) Feature vectors are computed for each segment to classify; (2) The feature vectors are classified using an adaptive distance; (3) Segment models are extracted from the computed feature classes. Similarities between segments inside a given class are expressed as links between feature vectors formalizing their "common fate".

mum. This feature vector constitutes an enriched description of a segment (step 1 in the Fig. 4).

A Euclidean distance seems an appropriate choice to compare these feature vectors, at least in this preliminary step of the design. Adaptive weights are used to model the mutual dependencies between classes and features, thus reflecting the fact that certain features play a more or less important role in class characterization. The weights are computed using a classical feature selection algorithm [27].

The choice of K-mean algorithm for the classification step was motivated by several requirements:

– *Regarding knowledge discovery:* The ability to run with a minimum of *a priori* knowledge (in particular no *a priori* knowledge of the number of classes to observe).
– *Regarding interaction:* The ability to support an incremental user-driven modification of a segment class without affecting the classification of other segments.
– *Regarding adaptation:* The ability to adjust the comparison criterion to cope with class specificities. An extended version of the K-mean algorithm is exploited to this end. In this variant, different distances (e.g. weights in the Euclidean distance computation) are selected and computed depending on the selected class.

A model of segment (step 3, Fig. 4) and a symbolic value are finally attached to each class. The next issue is to match this symbolic value with an interpretable denomination for the clinician. This is performed thanks to the interaction between man and machine. Several cases may occur in this respect:

– The clinician may provide a denomination in his own language, thus modifying the annotation at hand.
– The clinician may prefer to formalize segment-to-segment associations in case the interpretation is uncertain. These associations will be kept by the systems as ‘‘common fate’’ constraints in further processing.
– The clinician may ask the system to provide further examples of a given class model when neither of these situations holds, to look for an increased experimental assessment of the modeling hypothesis at hand.

4.2.3. Symbolic translation agent

The role of the symbolic translation agent is to translate numerical time series into symbolic time series. There are three reasons to independently perform symbolic translation and classification labeling. Firstly, a theoretical reason: to clearly separate the models construction task performed by classification agents from the use of these models performed by segmentation agents to describe the data. Secondly, a design related reason: since segmentation and classification agents work in an asynchronous way, it may happen that time series partitioning evolves while models are being constructed. And finally, a practical reason: to enable modification or creation of models by the user at the symbolic translation stage and consequently override segmentation and classification tasks.

Then, the role of the symbolic translation agent is to build a symbolic description that is situated in the context of the present analysis. For this purpose, it collects over all series types (T) the segments (S) and models of segments (M) computed by the segmentation and classification agents, and proceeds to their matching. Time-stamped symbols are then created for each segment in the form (S_{date}, S_{duration}, T, M). Symbolic time series are finally built by concatenating the time-stamped symbols for each series type.

Similar to the time series segmentation result, symbolic time series are visible via the interface. Each time-stamped symbol is a symbolic annotation of a time series that focuses the clinician attention on interesting parts of a record. The clinician may as well modify the attached symbols. As a matter of fact, modifying the date or duration of a symbol may be seen as equivalent to the modification of the corresponding segment date and/or duration. A modification of the symbol itself is equivalent to the attribution of a new class to the corresponding segment.

4.2.4. Scenario agents

In a way similar to data segmentation (finding relations between signal elements) and classification (finding relations between segment descriptors), scenario construction is considered as the process of finding temporal relations between symbols. To avoid the combinatorial explosion resulting from a unsupervised search, the process is driven by the assumption of some specific symbol that has to be ‘‘explained’’, by its causal links to some ancestor ‘‘events’’ (symbols).

Given a symbol E, pointed out by the clinician, the task of the scenario agent is then to:

1. Find all occurrences of E in the available symbolic time series.
2. Collect the symbols preceding each occurrence of E in a temporal window, focusing on a fixed length of this set of symbols. Each set of symbols in a temporal window constitutes a scenario instance. The date of E is considered as a time reference and all symbols are aligned to this reference.
3. Construct the time-stamped pattern corresponding to these scenario examples.
4. Construct the scenario that explains E.

The main step of the scenario construction consists in learning time-stamped patterns in symbolic time series. The A Priori algorithm [28] finds the largest frequent pattern in a sequence of symbols. To learn the scenario, we use an extended version of this algorithm [29] that builds the largest frequent time-stamped pattern. Other temporal extension of the A Priori algorithm can be found in [30,31]. This algorithm learns quantitative temporal constraints (date and duration of symbols) from symbolic time series examples. For Dousson [29], the pattern frequency is calculated from the number of matches in a given symbolic time series. In our case, it is computed as the frequency of examples containing a matching set of symbols in the whole example collection.

Scenarios are constructed as the result of this process, which "explains" the event of interest E by a time-stamped pattern of preceding events. This result may of course be submitted to the interacting user, which may result in modifications in the proposed pattern, or the collection of other example patterns to support or contradict the current interpretation.

In a complementary feedback loop (link 8 in Fig. 2), the proposed scenario may be considered as a model to drive further analysis. As a matter of fact the A Priori algorithm also computes the most frequent sub-scenarios while computing scenarios. Then, given a scenario of which only some of its sub-scenarios appear to match a given time series data, a deviation from the current model is pointed out, and therefore a potential improvement of the annotation process is possible. The system will then focus its attention on locations in the available symbolic time series where a deviation from this model is observed. Feedback toward the lower analysis levels may then occur in two ways, depending whether the issue is to revise the proposed classification (symbolic assignment, link 4, Fig. 2) or the proposed segmentation (temporal assignment, link 3, Fig. 2).

5. Experiments

A standard frame for evaluation of collaborative systems is presently not available due to the difficulties of modeling and classifying the large diversity of the existing systems [32]. The central question is: Does the collaboration enable the user or the system to perform "better" together than they could do independently? To answer this question two aspects have to be carefully considered: the global *performance* and the *usability* (adequacy with user requirements) of the approach. For the evaluation of these two aspects, Shyr et al. [33] have proposed a framework and guidelines oriented toward three dimensions:

– The stage and place of the experiment: laboratory evaluation or field experiments.
– The quality of the system: performances and usability.
– The evaluation methodology used: with objective (via quantitative measures) or subjective (via user feedbacks) benchmarks.

We have refined and applied this methodology to our context.

5.1. Our evaluation methodology

For our application, laboratory experiments consider feedback from researchers of our group (including physiologists and clinicians) exploring the real data with our tool outside the usual clinical environment. Field experiments consist of the evaluation of the use of the tool by a clinician at the patient's bedside. The usability evaluation includes human–machine interaction (HMI) criteria (knowledge sharing efficiency, HMI quality, etc.) and cognitive science criteria (work load evaluation, result confidence, etc.). It requires the use of interviews and questionnaires to gather experiment feedbacks. In term of acceptability by the users, the adequacy of the collaboration in the clinical environment and gains compared to the standard practice should be evaluated via clinical trials.

For time series data exploration, the system performances can be assessed considering three aspects:

– The capability to efficiently explore and annotate a large amount of time series data.
– The quality of the built computerized models of events and scenarios.
– The capability of new events and scenarios discovery.

With simulated data sets, objective measures of quality can be proposed. Then, we can compare performances between a fully, partial or absent collaboration with various systems.

In addition, because we propose a new approach for collaboration, we should evaluate its feasibility. Feasibility means that our implemented system: (1) provides an effective structural coupling collaboration, preserving the autonomy and adaptability properties of the system. This includes the capabilities for model construction, where models emergence and automatic annotation should be highlighted; and (2) exhibits specific characteristics in comparison to other collaborative approaches.

Our evaluation framework based on three levels: feasibility, performance, and usability is summarized in Table 2.

In this paper, we present the results we obtained for feasibility and performance evaluation from laboratory experiments. Based on our methodology (see Table 2) three aspects were considered:

– Collaborative specificity compared to other systems ones: we explored physiological time series with two relevant tools, VizTree [10] and TimeSearcher2 [12] in order to define their limits and their complementary with our approach (Section 5.3).
– Expected properties verification: we tested on real medical time series data the expected properties of our collaborative system in terms of its autonomy (e.g. is the system able to learn and then perform automatic annotations?), emergence (*e.g.* is new knowledge co-constructed based on man–machine collaboration?) and finally adaptability (e.g. is the system able to cope with patterns variability and heterogeneity of features associated to each class?) (Section 5.4).

Table 2
Experimental evaluation methodology adapted from Shyr et al. [33]

	Laboratory experiments Participant: Research team including clinicians Time series data: Simulated or real data Tool: Laboratory prototype	Field experiments Participants: Clinicians at the patient's bedside Time series data: Real data Tool: Clinical prototype
Feasibility evaluation	*Criteria* – Expected properties verification – Collaborative specificity compared to other systems ones *Objective methods* – Existing time series exploration system comparisons *Subjective methods* – Interview with potential users (Is the system interesting to explore time series data?)	
Performance evaluation	*Criteria* – Automatic annotation reliability and model construction efficiency on large data sets – Specificities and sensitivities of events and scenarios models *Objective methods* – Results analysis on simulated time series – Results analysis on real time series	*Criteria* – Impact on clinical care *Objective methods* – Clinical trial
Usability evaluation		*Criteria* – Collaboration acceptance (system annotation relevance, results confidence) – Collaboration effectiveness (time saving, cognitive load reducing) – HMI usability (annotation interfaces and visual feedbacks) *Objective methods* – User activity trace analysis – Learning time, tasks execution time – Error rate *Subjective methods* – System comparisons with different levels of collaboration – Interview, Questionnaire – User activity observation

– Specificities and sensitivities of events models are presented in Section 5.4.3.

5.2. Rationale for patient-ventilator asynchronies exploration

Patients suffering from respiratory disorders and hospitalized in intensive care units are mechanically assisted with a ventilator. One main objective of using assisted or patient-triggered mechanical ventilation is to avoid ventilator-induced diaphragmatic dysfunction, by allowing the patient to generate spontaneous efforts. An adequate synchronization between the patient and the ventilator is likely to improve patient's comfort and optimize work of breathing [34]. Patient ventilator asynchronies can be defined as a mismatch between patient and ventilator inspiratory and expiratory times. The incidence of major patient-ventilator asynchronies during mechanical ventilation is poorly known. A recent study [35] shows that 24% of 60 patients mechanically ventilated exhibited an asynchrony index higher than 10% of respiratory efforts. Asynchrony was associated with a longer duration of mechanical ventilation. Ineffective triggering, i.e. when patient's efforts do not result in ventilator triggering was pointed out as the major cause (85%) of asynchrony. Their detection would imply a time-consuming careful exploration of respiratory recordings by experienced clinicians. To limit asynchronies, mechanical respiratory support should be continuously adapted to follow the evolution of the patient's needs in particular during sleep. In practice, this adaptation can not be performed by the clinical staff. This motivates the design of computerized assistants that continuously adapt the assistance when for instance asynchronies are detected. Therefore, it is important (1) to automatically detect asynchrony, (2) to identify factors increasing the incidence of asynchrony, and (3) to automatically optimize ventilatory settings and then minimize mismatch between the patient and the ventilator.

In order to detect from the physiological data the specific patterns regularities or sequences of events (scenarios) associated to the occurrence of asynchrony, we provided a clinician with our system to annotate and explore the mass of data available. Times series were constituted of flow and airway pressure (P_{aw}) signals continuously recorded during 30 min and sampled at 200 Hz. Details on the data acquisition protocol may be found in [35]. Only the most stable part of the record is exploited. Each time series contained about 64,000 data points, i.e. around 100 respiratory cycles. Fig. 7 shows only few respiratory cycles of the respiratory flow signal.

A clinician, based on a visual inspection of three patients recordings, manually annotated ineffective triggering asynchronies by identifying, during the expiratory periods, depressions, defined as an abrupt airway pressure drop ($\geqslant 0.5$ cm H_2O), simultaneous to an increase in flow not followed by an assisted cycle.

5.3. Time series exploration with other existing collaborative approaches

We have explored the two following tools, VizTree [10] and TimeSearcher2 [12], which are similar to our approach, our respiratory multivariate time series data. Note that the same flow signal part is shown on Figs. 5–7.

5.3.1. VizTree

VizTree processes a single univariate time series data. The exploration is centered on the pattern discovery and detection tasks. Any relation between patterns can be explored. The underlying model of pattern is a symbolic representation based on SAX [11]. The tool proposes three simultaneous graphical views of the data:

– A signal-based representation of the time series at hand, which allows to highlight the segments matching the patterns under interest.
– A tree-like representation organizing the time series as a hierarchy of patterns. Patterns may be selected from this tree, at any level. eventually according to some advanced selection criteria (e.g. non-monotonic string), and
– A cumulative representation of the segments matching the selected pattern that enables the user to visually check the homogeneity of the segments and therefore the ‘‘representativity’’ of the pattern.

The visual support provided by VizTree to explore time series data is attractive (see Fig. 5). The user can quickly perform a lot of experiments, exploring the time series in a wide range of patterns. Data interpretation results from a series of back and forth steps between the interesting branch of the tree and the cumulative view of segments. The detection of the length of the pattern is *a priori* fixed by the expert. This is clearly a limitation for medical applications where similar segments can have variable temporal scales. Moreover, the exploration remains user-guided. There is no pattern discovery tool enabling the system to autonomously exhibit potentially interesting findings from the data.

5.3.2. TimeSearcher2

TimeSearcher2 is a multivariate time series data exploration tool including the notion of SeachBoxes. SearchBoxes enable users to select a subset of an existing time series and search for similar patterns throughout the data. Various controls over the acceptable similarities are provided. Several facilities have been added to the early version of the system, involving time varying boxes, or angular queries which search for range of differentials rather than absolute values. Fig. 6 shows an experimental use of SearchBox on our patient data. The quality of the pattern detection (Fig. 6a) is very sensitive to the threshold fixed by the user (here 48%).

The difficulty of the system to cope with patterns heterogeneity (lack of adaptability) limits its capacity of detection of similar patterns in different flow signals (Fig. 6b). TimeBox is another appropriate tool available in TimeSearcher2. It enables the user to select example patterns by means of adjustable rectangular boxes. Multiple TimeBoxes can be drawn to specify conjunctive or disjunctive queries of arbitrary complexity. However, TimeBoxes are presently not adapted to deal with long periodic time series data and then are not useful for our application where cyclic patterns should be detected.

5.4. Testing our system properties

The three following subsections are respectively devoted to testing the autonomy and adaptability (Section 5.4.1) and emergence (Sections 5.4.2, and 5.4.3) properties of our system.

5.4.1. Model construction and automatic annotations

Using the system interface, the clinician annotated inefficient triggering (see Fig. 7a) by the visual inspection of a specific part of the signal. He was then able to launch the processing, i.e. in turn the execution of the classification, segmentation and finally symbolic translation agents. Based on the initial partial annotation by the expert, the system symbolically labeled the complete time series (see Fig. 7b). On this example, all asynchronies considered as similar to those annotated by the clinician as ineffective triggering were retrieved (dark gray boxes). Standard cycles (no asynchrony) were automatically segmented (light gray boxes). These ambiguous cases will be resolved by means of further clinician interactions.

5.4.2. Emergence of new models

As may be seen from Fig. 8, three different models have been constructed by the system. The first model corresponds to ineffective triggering, according to the expert's annotations. The second model on the contrary is representative of non asynchrony periods. The third model is pro-

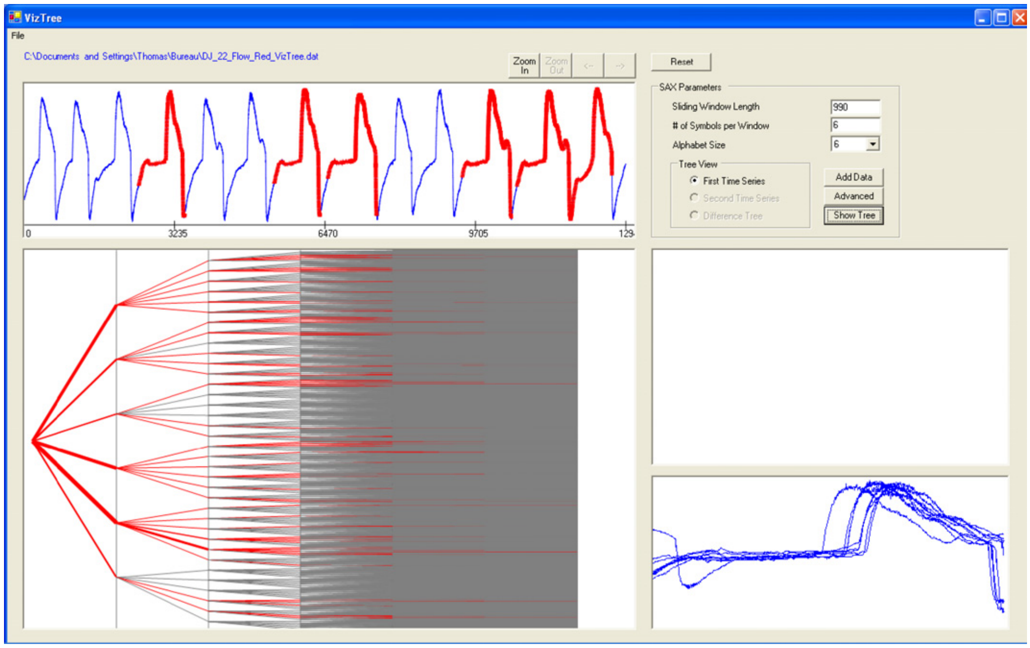

Fig. 5. Exploration of Flow signal using VizTree: Top in red are shown the patterns detected by the system based on the first one (left) annotated by the clinician. Note that due the use of a fixed window for pattern detection, some parts can be missed when the pattern is longer (see the third pattern). Below left: Tree like organization of the times series. Below right: The superposition of the patterns detected. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

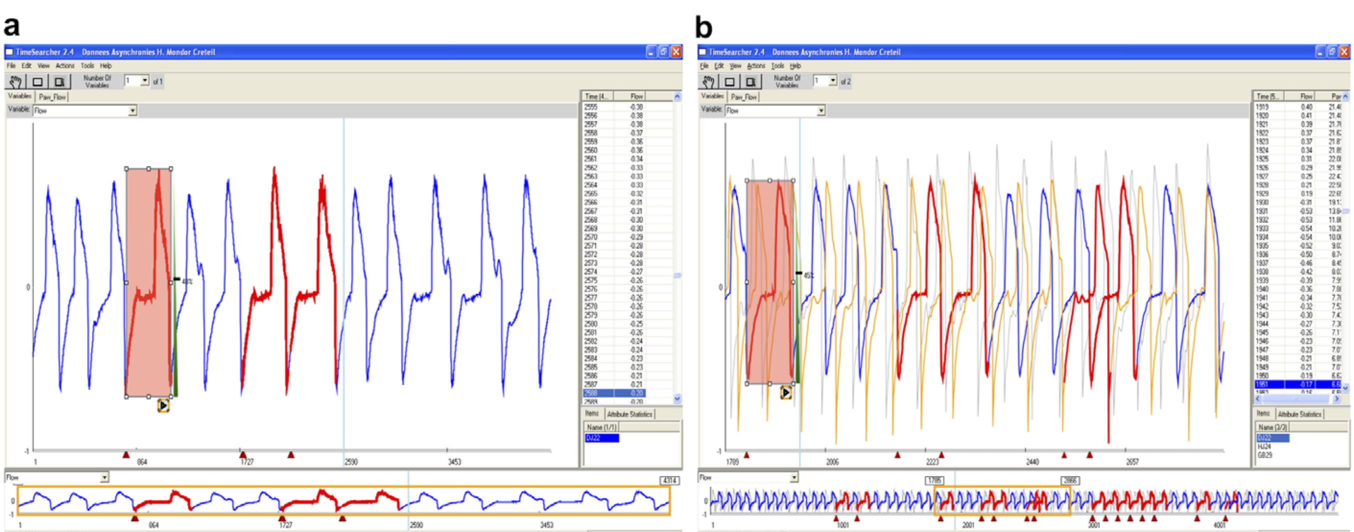

Fig. 6. Exploration flow signal under TimeSearcher2. (a, left): The green box indicates the example pattern to detect, in red the corresponding detected patterns. (b, right): Excerpt of the three flow signals available from three patient recordings. The patterns (in red) learned on the blue signal are fixed and cannot be adapted to detect similar patterns on the two other signals. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

posed by the system, in an emergent way, to cope with a specific case of asynchrony. The clinician at this step may reject the model or proceed to further time series exploration to investigate the genericity of this newly defined model.

5.4.3. Comparison of models

Pathophysiological aspects of asynchrony are not well known. It is therefore interesting for clinicians to explore the patterns of asynchrony cycles provided by the system.

We used our system to investigate whether these models are patient-dependent or whether a generic model can be extracted. To illustrate this point, we used data obtained from three patients (identified by DJ22, GB27, and HJ24). Two signals, *Paw* and *Flow*, were firstly annotated by the clinician. The model of ineffective triggering was automatically constructed from the data of one patient and then used to annotate the recordings of the two other patients. By comparing the results obtained by the machine

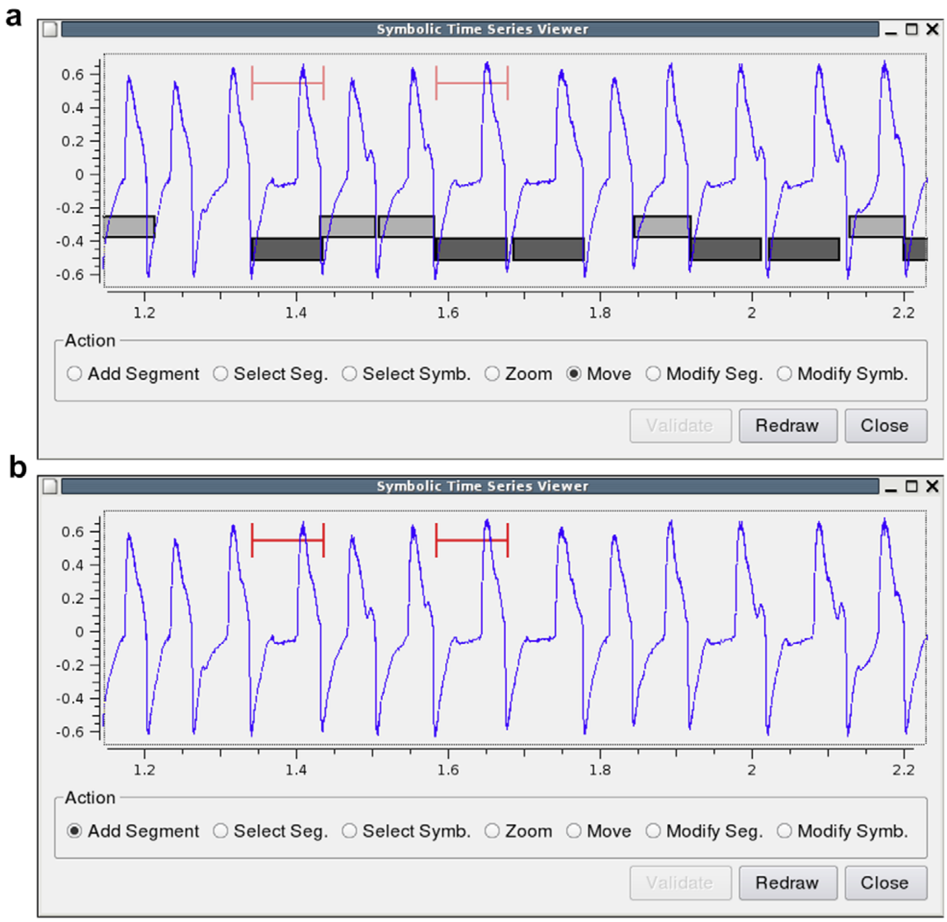

Fig. 7. Model construction (flow signals). (a, upper): Two annotations (red horizontal bar) are inserted on the flow signal by the clinician to indicate asynchrony periods. (b, lower): These annotations are used by the system to symbolically label the complete time series: dark-gray boxes indicate retrieved asynchronies periods. Other gray boxes indicate other types of periods: non-asynchronies. Note the variability of the patterns gathered in the same class (e.g. light-gray boxes). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

with the expert annotations, we calculated the sensitivity and specificity of the constructed model. The results are presented in Tables 3 and 4, respectively.

The mean sensitivity and mean specificity of our approach, computed over the three patients, is respectively 0.75 and 0.95. We can note that for the models learned from DJ22 data provide systematically lower specificity and sensitivity than those obtained from GB27 and HJ24. This can be explained by the fact that fewer asynchronies occurred within DJ22 time series data (24% less than in GB27 and HJ24). The importance of the number of available annotations is illustrated by the ROC curve (see Fig. 9). For each point, sensitivity and specificity were computed on 20 randomized trials. The area under the curve (AUC) is equal to 0.88.

6. Discussion and perspectives

We have presented an original approach to support clinicians in the difficult task of data exploration for knowledge construction from multivariate time series data. Our approach is centered on the collaboration between a clinician and an autonomous system. Man and machine are both embedded in a learning cycle. This extends the expert's involvement as already proposed in [1] to the entire knowledge discovery cycle. We advocate that the design of computerized tools fully support the clinician in his/her making decision process, rather than provide him/her with final results. The presented system engenders medical computerized tools that are designed on that basis [3].

The system has been designed as a recursive triadic architecture that is implemented with a multi-agent paradigm. Three types of agents have been introduced: segmentation agents that use model of patterns to find new similar segments, classification agents that build models of patterns from previously constructed segmentations, and finally, scenario agents that build typical relations between time-stamped symbols. We are aware of the computing complexity of the processes that we have presented here. This can be lowered by the fact that data interpretation is a prospective task with no critical time constraints. Moreover, new hardware such as grid architecture could be used to implement our system.

The proposed evaluation methodology includes three evaluation stages: approach feasibility, performance, and

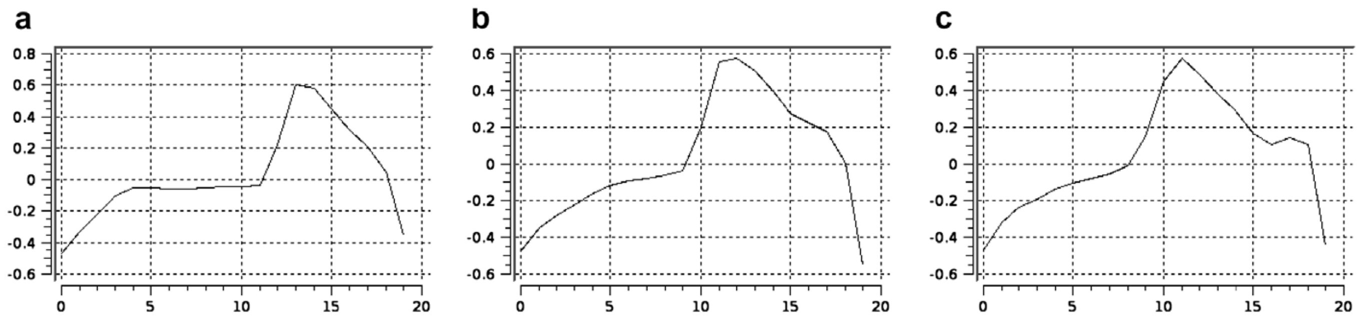

Fig. 8. Emergence of new models on flow signals (20 points). From the left to the right, (a) Asynchrony model with a large steady time, (b) Most common non-asynchrony model, (c) Discovered non-asynchrony with an ending artifact.

Table 3
Sensitivity values obtained for the detection of ineffective triggering for three different patients

Data model	DJ22	HJ24	GB27
DJ22	0.57	0.71	0.54
HJ24	0.82	0.80	0.50
GB27	0.68	0.72	0.71

For the *n*th raw, the *n*th patient (DJ22, HJ24 or GB27) was used as training set.

Table 4
Specificity values obtained for the detection of ineffective triggering, for three different patients

Data model	DJ22	HJ24	GB27
DJ22	0.87	0.98	0.85
HJ24	0.88	0.95	0.89
GB27	0.98	0.98	0.97

For the *n*th raw, the *n*th patient (DJ22, HJ24 or GB27) was used as training set.

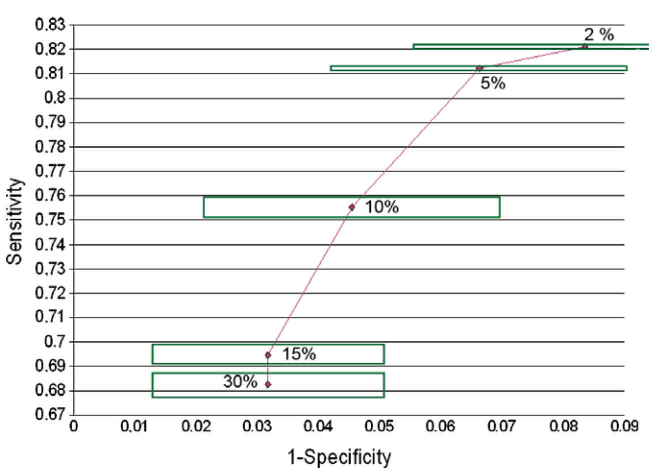

Fig. 9. ROC curve. Flow time series was manually annotated from 2% to 30% of the total asynchrony events. For each point, sensitivity and specificity were computed on 20 randomized trials. Boxes indicate the corresponding standard deviations (SD): height and width corresponds respectively to the SD of sensitivity and specificity.

usability of the system. Results obtained indicate the feasibility of ‘‘structural coupling’’ collaboration. This clearly distinguishes our system from the other collaborative systems described in the literature.

The laboratory experiments on patient ventilator asynchrony exploration show that our approach exhibits acceptable levels of sensitivity and specificity. These performances could be improved with a more complete system including scenario learning and feedback from scenario level to the event detection level. Clearly, our experiments are presently limited. However, they indicate that our approach has potential for generic model extraction. We have looked for patterns relevant to asynchrony and this can be applied to the detection of other patterns in time series data. Our current work aims at extending the system with an enriched description of segments, to fuse several physiological parameters for the construction of more elaborate asynchrony model (e.g. using some of the 306 different features to summary time segments that used in the data-driven temporal abstraction in [2]). We have considered time as a specific constraint in our pattern matching algorithm. Preliminary experiments on scenario learning indicate that a more sophisticated time representation should be introduced. For this purpose, formal logic can be considered (see [36] for an application to ECG interpretation) or an extension to the A Priori algorithm that we have recently proposed.

Based on the indications provided by our expert, only parts of the signal intrinsically consistent were used for clustering. In a similar way, data from various patients were carefully grouped. Similarly to KNAVE-II [9], a representation of the clinical context should be introduced in a future extension of the system to take into account of contextual information such as pathology or the mode of ventilation used.

We have reported results obtained with VizTree and TimeSearcher2, which integrate knowledge discovery tools to support in-depth exploration of time series. Despite of their visual interactive facilities to explore time series data, these tools are not really collaborative in a sense that the machine does not construct data interpretation which is refined via man–machine interaction.

The annotation paradigm was initially proposed in a clinical context with TSNet tool. TSNet [37] is based on a client–server architecture and provides an expandable range of facilities for viewing, annotating and analysing complex multi-channel time series data. However, in TSNet annotations provided by user and machine are seen as processing steps and are not to be used for a collaborative exploration issue. Manual annotation is time-consuming. This point motivates the use of an automatic annotation for a large part of the signal. Presently, the clinician annotates about 10% of each signal. We consider that this is inevitable and the only way to learn patterns. Nevertheless, this could be a limit to our approach. Further experimentations in close collaboration with clinicians will allow us to fully evaluate, using field experiments, the usability and performance of our approach in the context of ICU, and the real impact of such an interactive approach for knowledge discovery and knowledge formalization (see [1] for similar experiments) in the medical domain. Such evaluation should be quantitative and possibly include a way to measure the clinician's interactions with the machine.

Ideally, a computerized assistant for medical time series data exploration should mix several characteristics that we presently encounter in different recent systems: man–machine interaction facilities via powerful visualization tools (VizTree and TimeSearcher2), learning by examples capability (TimeBoxes in TimeSearcher2), and knowledge-based abstraction mechanisms (KNAVE) for scenario discovery. The collaborative system we propose is a first step towards the design of such an assistant.

Acknowledgments

T. Guyet is funded by the Centre National de la Recherche Scientifique (French Ministry of Research and Technology). The authors thank Dr. Arnaud Thille and Pr Laurent Brochard from Henri Mondor Hospital (Créteil, France) for providing the clinical data and for comments on this study, the anonymous reviewers for their many valuable suggestions, J. Bourbeillon and M. Jungmann for assisting with the last versions of the manuscript.

References

[1] Morik K, Imhoff M, Brockhausen P, Joachims T, Gather U. Knowledge discovery and knowledge validation in intensive care. Artif Intell Med 2000;19(3):225–49.

[2] Verduijn M, Dagliati A, Sacchi L, Peek N, Bellazzi R, de Jonge E, et al. Comparison of two temporal abstraction procedures: a case study in prediction from monitoring data. In: Friedman C, editor. Proceedings of AMIA Annual Symposium. Am Med Inform Assoc; 2005. p. 749–53.

[3] Zupan B, Holmes JH, Bellazzi R. Knowledge based data analysis and interpretation. Artif Intell Med 2006;37(1):163–5.

[4] Morik K. Balanced cooperative modeling. In: Michalski R, Tecuci, editors. Machine learning: a multistrategy approach. Morgan Kaufmann; 1994. p. 295–318.

[5] Schroeder LF Bazzan ALC. A multi-agent system to facilitate knowledge discovery: an application to bioinformatics. In ECCB: European Conference on Computational Biology and in Bioinformatics. 2002; p. 18:35–43.

[6] Berry L, Munzner T. BinX: dynamic exploration of time series datasets across aggregation levels. In: Ward M, Munzner T, editors. Proc IEEE Symp Inform Vis. INFOVIS'04. IEEE Computer Society; 2004. p. 215.

[7] Wattenberg M. Sketching a graph to query a time-series database. In Proc 2001 Conf Human Factors Computing Systems, Extended Abstracts; ACM Press. 2001; p. 381–82.

[8] Notsu H, Okada Y, Akaishi M, Niijima K. Time-tunnel: visual analysis tool for time-series numerical data and its extension toward parallel coordinates. In Proc Int Conf Computer Graphics, Imaging Vis, CGIV'05; IEEE Computer Society. 2005. p. 167–72.

[9] Shahar Y, Goren-Bar D, Boaz D, Tahan G. Distributed, intelligent, interactive visualization and exploration of time-oriented clinical data and their abstractions. Artif Intell Med 2006;38(2):115–35.

[10] Lin J, Keogh E, Lonardi S, Lankford J, Nystrom D. VizTree: a tool for visually mining and monitoring massive time series databases. In: Nascimento M, Özsu T, Kossmann D, Miller R, Blakeley J, Schiefer B, editors. Proc Thirtieth Int Conf Very Large Data Bases. VLDB'04. Morgan Kaufmann; 2004. p. 1269–72.

[11] Lin J, Keogh E, Wei L, Lonardi S. Experiencing SAX: a novel symbolic representation of time series. J Data Min Knowl Disc 2007. To Appear.

[12] Aris A, Shneiderman B, Plaisant C, Shmueli G, Jank W. Representing unevenly-spaced time series data for visualization and interactive exploration. In: Costabile MF, Paternò F, editors. Proc IFIP TC13 Int Conf. Human–Computer Interaction—INTERACT. Berlin/Heidelberg: Springer; 2005. p. 835–46.

[13] Augusto JC. Temporal reasoning for decision support in medicine. J Artif Intell Med 2005;33(2):1–24.

[14] Shahar Y. A framework for knowledge-based temporal abstraction. Artif Intell 1997;90(1–2):79–133.

[15] Black J, Robertson S, Zachary W. Cognition, computing, and cooperation. Intellect Edition 1990.

[16] Clarke A, Smyth M. A co-operative computer based on the principles of human co-operation. J Man-Mach St 1993;38(1):3–22.

[17] Salembier P. Etude Empirique et Modélisation d'une Activité de Diagnostic Cognitif. Intellectica. 1992;15:55–96.

[18] Horvitz E. Principles of mixed-initiative user interfaces. In: Proc CHI'99, ACM SIGCHI Conf Human Factors Computing Systems. 1999; p. 159–66.

[19] Amant St R, Cohen P. Interaction with a mixed-initiative system for exploratory data analysis. J Knowledge-based Sys 1998;10(5):265–73.

[20] Maturana HR, Varela FJ. Autopoiesis and cognition. Dordrecht. Holland: Reidel D, 1980.

[21] Maturana HR. Biology of language: the epistemology of reality. In: Miller GA, Lenneberg E, editors. Psychology and biology of language and thought: essays in Honor of Eric Lenneberg. New York: Academic Press; 1978. p. 27–63. Chapter 2.

[22] Steels L, Kaplan F. Collective learning and semiotic dynamics. Advances in artificial life. In Floreano D, Nicoud J-D, Mondada F editors. ECAL'99: Proc Fifth European Conf Artificial Life. 1999; p. 679–88.

[23] Stuber A, Hassas S, Mille A. Language games for meaning negotiation between human and computer agents. In: Dikenelli O, Gleizes MP, Ricci A, editors. ESAW'05: Proc Eng Soc Agents. -World; 2005. p. 275–87.

[24] Subrahmanian E, Monarch I. Annotations and Collaboration: One in Service of the Other. In Boujut J editor: Proceedings of the International Workshop on Annotation for Collaboration—Methods, Tools and Practices; CNRS—Programme Société de l'Information; 2005; p. 73–82.

[25] Steels L. Evolving grounded communication for robots. Trends Cogn Sci 2003;7(7):308–12.

[26] Schwarz E. Can real life complex system be interpreted with the usual dualist physicalist Epistemology—or is a holistic approach necessary? Res-Systemica, vol. 2 Special Issues Proc fifth Eur Sys Sci Congr, Crète, 2002.

[27] Molina LB, Belanche L, Nebot A. Feature selection algorithms: a survey and experimental evaluation. In Proc IEEE Int Conf Data Mining, ICDM 2002; IEEE Computer Society. 2002; p. 306–13.
[28] Agrawal R, Srikant R. Fast algorithms for mining association rules. In: Bocca JB, Jarke M, Zaniolo C, editors. Proc Twentieth Int Conf Very Large Databases. VLDB'94. Morgan Kaufmann; 1994. p. 487–99.
[29] Dousson C, Duong TV. Discovering chronicles with numerical time constraints from alarm logs for monitoring dynamic systems. In: Dean T, editor. Proc Int Joint Conf Artif Intell. IJCAI'99. Morgan Kaufmann; 1999. p. 620–6.
[30] Höppner F. Learning dependencies in multivariate time series. In Proc ECAI'02 Workshop Knowledge Discovery in (Spatio-)temporal Data. 2002; p. 25–31.
[31] Mannila H. Efficient algorithms for discovering association rules. In: Fayyad U, Uthurusamy R, editors. AAAI Workshop Knowledge Discovery Databases (SIGKDD). California: AAAI Press; 1994. p. 181–92.
[32] Jones P, Jacobs J. Cooperative problem solving in Human–machine systems: theory, models, and intelligent associate systems. IEEE Trans Systems, Man and Cybernetics, Part C: Appl and Rev 2000;30(4):397–407.
[33] Shyr P, Tecuci G, Boicu M. Evaluation of mixed-initiative knowledge base development methods and tools. In: Proc IJCAI-2001 Workshop on Empirical Methods in AI. 2001; p. 47–53.
[34] Sassoon CS, Forest GT. Patient-ventilator asynchrony. Curr Opin Crit Care 2001;7(1):28–33.
[35] Thille A, Rogriguez P, Cabello B, Lellouche F, Brochard L. Patient-ventilator asynchrony during assisted mechanical ventilation. J Intensive Care Med 2006;32(10):515–22.
[36] Fromont E, Quiniou R, Cordier MO. Learning rules from multi-source data for cardiac monitoring. In: Miksch S, Hunter J, Keravnou E, editors. Proc Tenth Conf Artif Intell Med. Berlin: Springer-Verlag; 2005. p. 484–93.
[37] Hunter J. TSNet. A distributed architecture for time series analysis. In Peek N, Combi C editors: Proc Workshop Intelligent Data Analysis Bio-Med Pharmacol, IDAMAP. 2006; p. 85–92.